ADA LAB 1BM21CS247

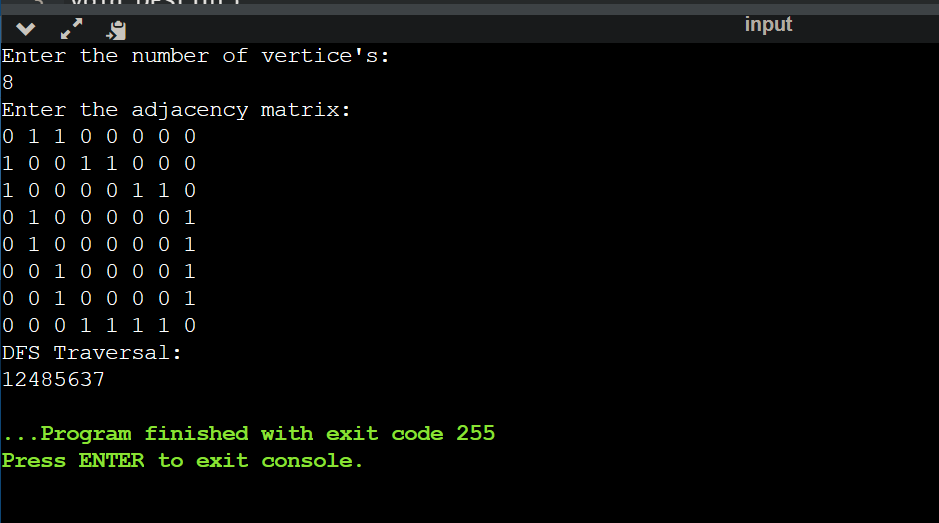
WEEK 2

**Q) DFS Traversal using C program**

**SOURCE CODE:**

#include<stdio.h>  
#include<conio.h>  
void DFS(int);  
int a[10][10], vis[10], n;  
void main()  
{  
    int i,j;  
    printf("Enter the number of vertice's:\n");  
    scanf("%d",&n);  
    printf("Enter the adjacency matrix:\n");  
    for(i=1;i<=n;i++)  
    {  
        for(j=1;j<=n;j++)  
        {  
            scanf("%d",&a[i][j]);  
        }  
    }  
    for(i=1;i<=n;i++)  
    {  
        vis[i] = 0;  
    }  
    printf("DFS Traversal:\n");  
    for(i=1;i<=n;i++)  
    {  
        if(vis[i] == 0)  
        {  
            DFS(i);  
        }  
    }  
    getch();  
}  
  
void DFS(int v)  
{  
    int i;  
    vis[v] = 1;  
    printf("%d",v);  
    for(i=1;i<=n;i++)  
    {  
        if((a[v][i] == 1) && (vis[i] == 0))  
        {  
            DFS(i);  
        }  
    }  
}

OUTPUT:



**Q) Check whether a given graph is connected or not using the DFS method.**

SOURCE CODE:

#include<stdio.h>  
#include<conio.h>  
int a[20][20],reach[20],n;  
void dfs(int v)  
{  
int i;  
reach[v]=1;  
for(i=1;i<=n;i++)  
if(a[v][i] && !reach[i])  
{  
printf("\n %d->%d",v,i);  
dfs(i);  
}  
}  
void main()  
{  
int i,j,count=0;  
  
printf("\n Enter number of vertices:");  
scanf("%d",&n);  
for(i=1;i<=n;i++)  
{  
reach[i]=0;  
for(j=1;j<=n;j++)  
a[i][j]=0;  
}  
printf("\n Enter the adjacency matrix:\n");  
for(i=1;i<=n;i++)  
for(j=1;j<=n;j++)  
scanf("%d",&a[i][j]);  
dfs(1);  
printf("\n");  
for(i=1;i<=n;i++)  
{  
if(reach[i])  
count++;  
}  
if(count==n)  
printf("\n Graph is connected");  
else  
printf("\n Graph is not connected");  
getch();  
}

OUTPUT:

